International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

Clustering mixed data using an Artificial Bee Colony

José F. Cabrera-Venegas, Yusbel Chávez-Castilla

Abstract*: In this paper, we have proposed a clustering technique which optimizes the total compactness and separation (measured using the Silhouette index) of the clusters. The proposed algorithm uses an Artificial Bee Colony (ABC) based optimization method as the underlying optimization criterion. We used similarity based prototypes as cluster centers. The proposed clustering technique is able to suitably handle mixed and incomplete data types in such a way that the original characteristics of the data are preserved. Assignment of points to different clusters is done based on a dissimilarity function rather than the Euclidean distance. Results on real-life data sets show that the proposed technique is well-suited to detect true partitioning from data sets. Results are compared with those obtained by four existing clustering techniques, one genetic algorithm based clustering technique (AGKA), the k-Prototypes (KP) algorithm, well-known based K-means clustering technique for similarity functions (KMSF) and a newly developed algorithm with dissimilarity based clustering technique (AD2011).

Keywords* : swarm intelligence, artificial bee colony, clustering, mixed data.

I. INTRODUCTION

Clustering is one of the major tasks in Machine Learning and Pattern Recognition. It is devoted to finding a natural structure in a data set. Unlike its supervised counterpart [1-6], clustering lacks of information about class labels, or any other predefined distribution of objects in a particular data set. Clustering algorithms use object descriptions and objects dissimilarities in order to group together very similar objects, in a manner that it accomplishes the maxima of "higher cohesion and low coupling", meaning that within-group similarity must be as high as possible, and between-group similarity as low as possible [7].

Clustering techniques are needed nowadays due to the need to obtain several groups in some domains, however, there are huge challenges on the clustering process. Several domains have objects with different attribute types [8, 9].

Revised Manuscript Received on December 30, 2019.
***** Correspondence Author
José F. Cabrera-Venegas*, Computer Science Department, University of Ciego de Ávila, Cuba. E-mail: fci.unica@gmail.com
Yusbel Chávez-Castilla, Computer Science Department, University of Ciego de Ávila, Cuba. E-mail: fci.unica@gmail.com

For example, a customer description can include simultaneously attributes such as age (integer), sex (nominal), salary (real), educational degree (nominal), employed (Boolean), among others [10]. This type of object description is per say a challenge for any algorithm [11-13]. The lacking of a metric space makes impossible the definition of a sum operator and also the scalar multiplication [14-23]. In the same way, numeric attributes often have a large amount of values, each with low frequency. It makes frequency-based solutions developed for categorical data impracticable for numeric data.

In addition, the presence of missing values in objects descriptions makes it more complex for any classification procedure [24-27]. Taking into account that dependencies may occur among attributes, estimating missing values is not always a feasible solution. On the other hand, there are many types of missing values, each of them with particular characteristics. That's why some authors have considered the best solution for mixed datasets is to develop algorithms that are able to manage the absence of information, as well as mixed data types [28-30].

In our research, we addressed the issue of clustering mixed data types databases, also including absences of information. We consider a dataset $O=\{o_1,\ldots, o_n\}$ of objects. Each object description A(o) has an equal number of attributes, $A=\{ a_1,\ldots, a_d\}$, and $A_i(o_j)$ denotes the value of the i-th attribute in the j-th object. We do not impose any restriction to the nature of the attributes [31-35]. We also assume that a dissimilarity function D exists and is able to manage objects descriptions in terms of A. We consider clustering algorithms which obtain a partition $C=\{c_1,\ldots, c_k\}$ of the input objects O, so that no object belongs simultaneously to more than one group.

The article has the following structure: section 2 reviews some of the existing algorithms for clustering mixed and incomplete data. Section 3 explains the swarm intelligence model based on an Artificial Bee Colony. Section 4 explains the new Clustering algorithm based on Artificial Bees (CAB) algorithm. Section 5 includes numerical experiments in both synthetic and real – life datasets. Also, we offer conclusions and future work.

II. PREVIOUS WORKS

Although there is a huge amount of research in both pure numerical and pure categorical data clustering techniques, the number of algorithms able to deal with mixed type attributes is significantly lower. However, the scientific community has an increased interest in this particular area nowadays,

due to the nature of objects descriptions collected in many fields [8, 36].

Reviewing all existing clustering algorithms for mixed data types is in our criteria a never-ending task, due to the dispersion of information and the continuous research in the topic.

However, we aim at reviewing some of the more relevant of them.

A. Classic k-means and its variants

k-means algorithm is possible the first clustering algorithm in the literature. It is the base of the partition approach to clustering. This algorithm aims at obtaining clusters of spherical shape, been compact and separated. The algorithm works as follows: first, k centers are selected randomly. Then, each point is assigned to its most similar center, and centers are recalculated, using arithmetic mean [3, 37-39]. The algorithm continues repeating this process until no change is made to cluster centers.

Several authors have proposed modifications to the k-means algorithm to handle mixed and incomplete data. All of them include a redefinition of the distance function, as well as the cluster centers.

Perhaps the first modification of k-means for mixed and incomplete data is the k-prototypes (KP) algorithm [40]. The KP algorithm uses a new method to obtain cluster centers. Instead of the arithmetic mean, it uses the mode of nominal attributes, and maintains the mean for numerical attributes. In addition, it uses as dissimilarity function, with weights $\omega=\{\omega_1, \ldots, \omega_d\}$ for each attribute. By doing so, the KP algorithm is able to cluster mixed type data, but it does not handle missing data [10, 41-44].

Another modification of the k-means is the given by García-Serrano and Martínez-Trinidad in 1999 [45]. They propose the k-Means with Similarity Function (KMSF) algorithm. It defines as parameter a similarity function able to deal with mixed and incomplete data and also considers as cluster center the object that maximizes the overall similarity with respect to every object in the cluster [46, 47].

Another variation of the k-means is the one made by Ahmad and Dey [48]. Their proposal includes a novel dissimilarity function, with attribute weights. Such function considers co-occurrences of attribute values, and the relative frequencies of them in the similarity computation.

In addition, they redefine the notion of cluster center. Instead of been a point, the proposal of [48] use a cluster description as cluster center. It contains the mean of each numerical attribute, and a set of pairs (value, count) for each categorical attribute. Such pairs have the corresponding attribute value and the number of points in a cluster that have this value. In 2011, the same authors [49] proposed a modification of the 2007 algorithm. They do not give in the paper any name for the new method, so we will call it AD2011 (Ahmad and Dey proposal of 2011). The AD2011 algorithm discretizes numeric attributes before the clustering process, using the Equal Width Discretization procedure. It also includes in the dissimilarity function the contribution λ of each attribute to the cluster.

The AD2011 algorithm includes two user-defined parameters. The first is the γ parameter, included in the attribute contribution computation, having a suggested value of 20. And the second is the S parameter, included in the discretization procedure of numeric attributes, having a suggested value of 5.

B. Genetic Algorithm based clustering

One of the well-known optimization techniques in Computational Intelligence are Genetic Algorithms (GA). They offer a viable solution for several optimization and classification problems. In 2010, Roy and Sharma [50] developed a clustering algorithm for mixed and incomplete data, based on Genetic Algorithms. The AKGA method uses in the fitness function the dissimilarity proposed by Ahmad and Dey in 2007 [48]. It also uses the same cluster center definition. The GA codifies each cluster by using an array of integer data, having length equal to the number of data points. Each position of the array (gene) designates the cluster assigned to the point in that position [51-54].

The GA uses as mutation strategy the cluster changing. To do so, it assigns a point to its closest center, offering a quickly convergence. It also has an elitist survival strategy. Other bio-inspired algorithm used for clustering can be found in [55].

III. ARTIFICIAL BEE COLONY OPTIMIZATION

We are interested in the Artificial Bee Colony (ABC) optimization procedure proposed by Karaboga and Basturk [56] for numerical optimization. In the ABC algorithm, the colony of artificial bees contains three groups of bees: employed bees, onlookers and scouts. A bee waiting on the dance area for making decision to choose a food source is called an onlooker. A bee going to the food source visited by it previously is named an employed bee. A bee carrying out random search is called a scout.

Main steps of the ABC optimization procedure

1. Initialize.
2. *Repeat*
   - 2.1. Place the employed bees on the food sources in memory.
   - 2.2. Place the onlooker bees on the food sources in memory.
   - 2.3. Send the scouts to the search area for discovering new food sources.

   Until (requirements are met)

Fig. 1.ABC optimization procedure

In the ABC algorithm, a food source represents a possible solution of the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. At the first step, the ABC generates a randomly distributed initial population of solutions (food source positions).

After initialization, the population of the positions (solutions) is subjected to repeated cycles of the search processes of the employed bees, the onlooker bees and scout bees. An artificial employed or onlooker bee produces a modification on the position (solution) in its memory for finding a new food source and tests the nectar amount (fitness value) of the new source (new solution).

Provided that the nectar amount of the new source is higher than that of the previous one the bee memorizes the new position and forgets the old one. Otherwise it keeps the position of the previous one.

The food source whose nectar is abandoned by the bees is replaced with a new food source by the scouts. In the ABC algorithm this is simulated by replacing the abandoned solution with a new randomly produced one. In the ABC algorithm, if a position cannot be improved further through a predetermined number of cycles called limit, then that food source is assumed to be abandoned.

This model, proposed for numerical optimization, can be modified for clustering mixed and incomplete data.

IV. CLUSTERING MIXED DATA USING THE ABC MODEL

The ABC model includes several elements, such as a food source and nectar amount of a food source. It also has parameters as food sources count, generation count and a limit to abandon a food source. In our clustering model based on the ABC model, we consider a food source as a candidate clustering solution. We use an integer array of length equal to the data count, where each position of the array indicates the cluster designated to the point in that position.

2	3	2	1	1	2	3	3	1	2

Fig. 2. Example of a food source (candidate clustering). In this example, we have ten objects, and three groups. Each position of the array contains the cluster assigned to each object.

Also, our food source includes a set of cluster centers. Each cluster center is defined as the object in the cluster that minimized the overall dissimilarity with respect to every other object in the cluster. We assume the existence of a dissimilarity function d, able to deal with mixed and incomplete data types. Such dissimilarity functions can be found in [57].

$$\text{center}_i = \arg\min_{\substack{o,p\in c_i \\ o\neq p}} \{d(o),(p)\} \tag{1}$$

We also consider that clusters should be as compact and separated as possible, so we use as nectar amount of each food source (candidate clustering) the Silhouette index [58]. The silhouette is the average, over all clusters, of the silhouette width of their points. If x is an object in the cluster c_i and n_i is the number of objects in c_i, then the silhouette width of x is defined by the ratio:

$$S(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}} \tag{2}$$

where *a(x)* is the average dissimilarity between x and all other objects in c_i, and *b(x)* is the minimum of the average dissimilarities between x and the objects in the other clusters.

$$a(x) = \frac{1}{n_i - 1}\sum_{\substack{y\in c_i \\ y\neq x}} d(x,y)$$
$$b(x) = \min_{\substack{h=1..k \\ h\neq i}} \left\{ \frac{1}{n_h} \sum_{y\in c_h} d(x,y) \right\} \tag{3}$$

Finally, the global silhouette index is defined by

$$S = \frac{1}{k}\sum_{i=1}^{k} \frac{1}{n_i} \sum_{x\in c_i} S(x) \tag{4}$$

CAB algorithm

Parameters:
- B: Food sources count.
- S: scout bee count.
- I: number of generations (iterations).
- L: number of generations in which a food source is exhausted.

1. Initialize.
   - 1.1. Randomly generate B food sources (clustering solutions).
   - 1.2. Create B employed bees, and assign each of them to a food source.
   - 1.3. Create B onlookers bees, and assign each of them to a food source (*modeling a dancing area*).
   - 1.4. Create S scout bees.
2. *Repeat*
   - 2.1. For each employed bee
     - 2.1.1. If the current solution has not reached the limit L:
       - 2.1.1.1. Taste the nectar amount of its food source (*current nectar*), using the Silhouette index.
       - 2.1.1.2. Generate a new solution, by modifying its food source.
       - 2.1.1.3. Taste the nectar amount of the modified solution (*new nectar*).
       - 2.1.1.4. If (*new nectar* > *current nectar*), forget the current solution, replace it by the new one, and set the solution limit to zero.
       - 2.1.1.5. Else, increase the limit of the current solution.
     - 2.1.2. Else, abandon the solution.
   - 2.2. For each onlooker bee:
     - 2.2.1. If its closet food source *f* is available (not abandon):
       - 2.2.1.1. Taste the nectar amount of *f* (*current nectar*), using the Silhouette index.
       - 2.2.1.2. Generate a new solution, by modifying *f*.
       - 2.2.1.3. Taste the nectar amount of the modified solution (*new nectar*).
       - 2.2.1.4. If (*new nectar* > *current nectar*), forget the current solution, replace it by the new one, and set the solution limit to zero.
       - 2.2.1.5. Else, increase the limit of the current solution.
   - 2.3. For each scout bee:
     - 2.3.1. For each abandon food source:
       - 2.3.1.1. Replace it by a randomly generated new food source.
   - 2.4. Assign an employed bee to each new randomly generated food source.

   Until (the predefined number of generations I is reached).
3. Return the best food source.

Fig. 3.CAB algorithm

For a given object x, its silhouette index varies from −1 to 1. Values close to -1 means that the point is more similar, on average, to another cluster than the one to which it belongs. On the contrary, values close to 1 means that the average dissimilarity of the point to its own cluster is much more small than to any other cluster. The higher the silhouette, the more compact and separated are the clusters.

Our bees are modeled in the following: each bee has a food source, and has a method for producing a modification on the position (solution) in its memory and checking the nectar amount of the candidate source (solution). The modification of the food source (candidate clustering) is inspired in the mutation strategy of the AGKA algorithm [50].

We check every object in the candidate clustering (current food source), and assign it to its most probable cluster. We calculate the probability of every object *o* for each cluster $p_i(o)$, by taking into account the dissimilarity to the object with respect to every cluster center, and the higher dissimilarity (d_{MAX}) of the object with respect to a cluster center. In this way, we provide more compact and separated solutions.

$$p_i(o) = \frac{1.5 * d_{MAX}(o) - d(o, c_i) + 0.5}{\sum_{j=1}^{k} 1.5 * d_{MAX}(o) - d(o, c_i) + 0.5} \quad (5)$$

The bias 0.5 is introduced to avoid divide by zero error in the case that all patterns are equal and are assigned to the same cluster in the given solution.

Our new algorithm CAB optimizes the total compactness and separation (measured using the Silhouette index) of the clusters. The proposed algorithm is able to suitably handle mixed and incomplete data types in such a way that the original characteristics of the data are preserved.

V. EXPERIMENTAL RESULTS

In this research we use external evaluation measures for clustering. By suing such measures, we can determine if the clustering algorithm correctly assigns the points. In each cluster must be all and only those data objects that are assigned to the same class [59]. We conduct the experiments with 8 UCI labeled databases [60], with mixed and incomplete data. We used as external measures the Purity and Entropy indices, the F - Measure and the V-Measure, as described in [59].

Table- I: Dataset description

Datasets	Information		
	Categorical Attributes	*Numerical attributes*	*Classes*
autos	10	16	6
colic	15	7	2
dermat	1	33	6
heart-c	7	6	5
hepatitis	13	6	2
labor	6	8	2
lymph	15	3	4
tae	2	3	3

Setting algorithm parameters is often a difficult task [61-64]. Usually, several experimental results lead to an optimized value for the parameters. However, we do not optimize the parameters of our CAB method. In table II, we present the parameters used in our experimentation for the CAB method.

Table- II: Parameters of the CAB method used in the experiments

Parameter	Value
Food sources count	10
Scout bee count	1
Number of generations (iterations)	10
Number of generations in which a food source is exhausted	10

We compare the different clustering algorithms using each index. We also apply the Wilcoxon test to the results. By this, we obtain the statistical wins, losses and ties of each pair of algorithm. In table III, we offer the results of the experimental comparison. Table shows at each cell a triad reflecting the times our proposal outperforms, loses and ties with respect to reported methods, according to Wilcoxon test and the corresponding validity index.

Table III: Results of the comparison of the algorithm after Wilcoxon test

Validity Index	CAB versus			
	AD2011	*AKGA*	*KMSF*	*KP*
Entropy	8 - 0 - 0	7 -1- 0	4 - 0 - 4	5 - 3- 0
Purity	4 - 0 - 4	7 - 1- 0	3 - 4 - 1	3 - 2 - 3
F Measure	6 - 2 - 0	5 - 1 - 2	4 - 3 - 1	4 - 1 -3
V Measure	7 - 1 - 0	0 - 1 - 7	2 - 1 - 6	5 - 2 -1
Overall	25 - 3 - 4	19 - 4 - 9	13 - 8 - 12	17 - 8 - 7

As shown in table III, our proposal has a good performance according to the validity indexes used. According to Entropy, CAB loses once versus the genetic bases clustering AGKA, and three times versus the KP method. Similarly, according to Purity, our technique loses once versus AGKA, twice versus KP and in half databases versus KMSF. The F – Measure index shows a similar behavior. Our proposal loses twice versus AD2011, once versus AGKA and KP and three times versus KMSF. Finally, using the V- Measure, CAB loses twice versus KP, and only once versus other methods.

On average, the CAB method always outperforms all other algorithms, but in Purity with respect to the KMSF. It is remarkable that our proposal obtains these results with only 10 bees and in 10 generations, which is very quickly and with a low computational cost

VI. CONCLUSION

In this paper we propose a novel clustering technique based on Artificial Bee Colonies. We obtain compact and well – separated clusters, by maximizing the Silhouette index. Our proposal handles mixed and incomplete data types, and results on a good estimation of the true partitions of data. Our main contribution in this paper is to provide an algorithm framework for mixed and incomplete data types clustering, in which other codification strategies, fitness functions and modification strategies can be easily applied. In the future work, we will explore other alternative functions and strategies, as well as using other swarm intelligence models

REFERENCES

1. Medina-Pérez, M.A., et al. Selecting objects for ALVOT. in Iberoamerican Congress on Pattern Recognition. 2006. Springer.

2. Villuendas-Rey, Y., et al. Simultaneous features and objects selection for Mixed and Incomplete data. in Iberoamerican Congress on Pattern Recognition. 2006. Springer.
3. Rudas, I.J., et al. Generators of fuzzy operations for hardware implementation of fuzzy systems. in Mexican International Conference on Artificial Intelligence. 2008. Springer.
4. Villuendas-Rey, Y., M. García-Borroto, and J. Ruiz-Shulcloper. Selecting features and objects for mixed and incomplete data. in Iberoamerican Congress on Pattern Recognition. 2008. Springer.
5. García-Borroto, M., et al. Using maximum similarity graphs to edit nearest neighbor classifiers. in Iberoamerican Congress on Pattern Recognition. 2009. Springer.
6. García-Borroto, M., et al. Finding small consistent subset for the nearest neighbor classifier based on support graphs. in Iberoamerican Congress on Pattern Recognition. 2009. Springer.
7. Barroso, E., Y. Villuendas, and C. Yanez, Bio-inspired algorithms for improving mixed and incomplete data clustering. IEEE Latin America Transactions, 2018. **16**(8): p. 2248-2253.
8. Cerón-Figueroa, S., et al., Instance-based ontology matching for e-learning material using an associative pattern classifier. Computers in Human Behavior, 2017. **69**: p. 218-225.
9. López-Yáñez, I., et al., Collaborative learning in postgraduate level courses. Computers in Human Behavior, 2015. **51**: p. 938-944.
10. Moreno-Moreno, P., C. Yanez-Marquez, and O.A. Moreno-Franco, The new informatics technologies in education debate. International Journal of Technology Enhanced Learning, 2009. **1**(4): p. 327-341.
11. Villuendas-Rey, Y., et al., An Extension of the Gamma Associative Classifier for Dealing With Hybrid Data. IEEE Access, 2019. **7**: p. 64198-64205.
12. Villuendas-Rey, Y., et al., NACOD: A Naïve Associative Classifier for Online Data. IEEE Access, 2019. **7**: p. 117761-117767.
13. Villuendas-Rey, Y., et al., The naïve associative classifier (NAC): a novel, simple, transparent, and accurate classification model evaluated on financial data. Neurocomputing, 2017. **265**: p. 105-115.
14. Acevedo, M.E., C. Yáñez-Márquez, and M.A. Acevedo, Associative models for storing and retrieving concept lattices. Mathematical Problems in Engineering, 2010. **2010**.
15. Lopez, S.J., O.C. Nieto, and J.I.C. Oria, Non-parametric modeling of uncertain hyperbolic partial differential equations using pseudo-high order sliding mode observers. International Journal of Innovative Computing, Information and Control, 2012. **8**(3): p. 1501-1521.
16. Villuendas-Rey, Y., Y. Caballero-Mota, and M.M. García-Lorenzo. Using rough sets and maximum similarity graphs for nearest prototype classification. in Iberoamerican Congress on Pattern Recognition. 2012. Springer.
17. Villuendas-Rey, Y., Y. Caballero-Mota, and M.M. García-Lorenzo. Intelligent feature and instance selection to improve nearest neighbor classifiers. in Mexican International Conference on Artificial Intelligence. 2012. Springer.
18. Villuendas-Rey, Y., Y. Caballero-Mota, and M.M. García-Lorenzo. Prototype selection with compact sets and extended rough sets. in Ibero-American Conference on Artificial Intelligence. 2012. Springer.
19. Villuendas-Rey, Y., et al. Nearest prototype classification of special school families based on hierarchical compact sets clustering. in Ibero-American Conference on Artificial Intelligence. 2012. Springer.
20. Cleofas-Sánchez, L., et al. Hybrid associative memories for imbalanced data classification: an experimental study. in Mexican Conference on Pattern Recognition. 2013. Springer.
21. Villuendas-Rey, Y., M.M. Garcia-Lorenzo, and R. Bello. Support Rough Sets for decision-making. in Fourth International Workshop on Knowledge Discovery, Knowledge Management and Decision Support. 2013. Atlantis Press.
22. Yanez-Marquez, C., et al., BDD-based algorithm for the minimum spanning tree in wireless ad-hoc network routing. IEEE Latin America Transactions, 2013. **11**(1): p. 600-601.
23. Zavala, A.H., et al., Conjunction and disjunction operations for digital fuzzy hardware. Applied Soft Computing, 2013. **13**(7): p. 3248-3258.
24. Roman-Godinez, I., I. Lopez-Yanez, and C. Yanez-Marquez. A new classifier based on associative memories. in 2006 15th International Conference on Computing. 2006. IEEE.
25. Acevedo-Mosqueda, M.E., C. Yáñez-Márquez, and I. López-Yáñez, Alpha–Beta bidirectional associative memories: theory and applications. Neural Processing Letters, 2007. **26**(1): p. 1-40.
26. Guzman, E., et al. Image recognition processor based on morphological associative memories. in Electronics, Robotics and Automotive Mechanics Conference (CERMA 2007). 2007. IEEE.
27. Yáñez-Márquez, C., et al. Using alpha-beta associative memories to learn and recall RGB images. in International Symposium on Neural Networks. 2007. Springer.
28. Villuendas-Rey, Y., et al., Simultaneous instance and feature selection for improving prediction in special education data. Program, 2017. **51**(3): p. 278-297.
29. Serrano-Silva, Y.O., Y. Villuendas-Rey, and C. Yáñez-Márquez, Automatic feature weighting for improving financial Decision Support Systems. Decision Support Systems, 2018. **107**: p. 78-87.
30. Villuendas-Rey, Y., et al., Medical Diagnosis of Chronic Diseases Based on a Novel Computational Intelligence Algorithm. J. UCS, 2018. **24**(6): p. 775-796.
31. Ortiz-Ángeles, S., et al., Electoral Preferences Prediction of the YouGov Social Network Users Based on Computational Intelligence Algorithms. J. UCS, 2017. **23**(3): p. 304-326.
32. Ramírez-Rubio, R., et al., Pattern classification using smallest normalized difference associative memory. Pattern Recognition Letters, 2017. **93**: p. 104-112.
33. González-Patiño, D., Y. Villuendas-Rey, and A.J. Argüelles-Cruz, The potential use of bioinspired algorithms applied in the segmentation of mammograms. 2018.
34. Hernández-Castaño, J.A., et al., Experimental platform for intelligent computing (EPIC). Computación y Sistemas, 2018. **22**(1): p. 245-253.
35. Yáñez-Márquez, C., et al., Theoretical Foundations for the Alpha-Beta Associative Memories: 10 Years of Derived Extensions, Models, and Applications. Neural Processing Letters, 2018. **48**(2): p. 811-847.
36. Villuendas-Rey, Y., Maximal similarity granular rough sets for mixed and incomplete information systems. Soft Computing, 2019. **23**(13): p. 4617-4631.
37. Guzmán, E., et al., Morphological transform for image compression. EURASIP Journal on advances in signal processing, 2008. **2008**(1): p. 426580.
38. Moreno-Moreno, P. and C. Yáñez-Márquez. The new informatics technologies in education debate. in World Summit on Knowledge Society. 2008. Springer.
39. Yáñez-Márquez, C., I. López-Yáñez, and G.d.l.L.S. Morales. Analysis and prediction of air quality data with the gamma classifier. in Iberoamerican Congress on Pattern Recognition. 2008. Springer.
40. Huang, Z. Clustering large data sets with numeric and categorical values. in 1rst Pacific - Asia Conference on Knowledge discovery and Data Mining. 1997.
41. Godínez, I.R., I. López-Yánez, and C. Yánez-Márquez, Classifying patterns in bioinformatics databases by using Alpha-Beta associative memories, in Biomedical Data and Applications. 2009, Springer. p. 187-210.
42. Rudas, I.J., et al. Digital fuzzy parametric conjunctions for hardware implementation of fuzzy systems. in 2009 IEEE International Conference on Computational Cybernetics (ICCC). 2009. IEEE.
43. Zavala, A.H., et al. Parametric operations for digital hardware implementation of fuzzy systems. in Mexican International Conference on Artificial Intelligence. 2009. Springer.
44. Zavala, A.H., et al. VLSI Implementation of a Module for Realization of Basic t-norms on Fuzzy Hardware. in 2009 IEEE International Conference on Fuzzy Systems. 2009. IEEE.
45. García-Serrano, J.R. and J.F. Martínez-Trinidad. Extension to c-means algorithm for the use of similarity functions. in 3rd European Conference on Principles of Data Mining and Knowledge Discovery. 1999. Prague, Czeck Republic.
46. Aldape-Pérez, M., et al., Collaborative learning based on associative models: Application to pattern classification in medical datasets. Computers in Human Behavior, 2015. **51**: p. 771-779.
47. Guo-Hua, S., et al., Shannon information entropies for position-dependent mass Schrödinger problem with a hyperbolic well. Chinese Physics B, 2015. **24**(10): p. 100303.
48. Ahmad, A. and L. Dey, A k-means clustering algorithm for mixed numerical and categorical data. Data & Knowledge Engeneering, 2007. **63**: p. 503-527.
49. Ahmad, A. and L. Dey, A k-means type clustering algorithm for subspace clustering of mixed numeric and categorical data. Pattern Recognition Letters, 2011. **32**: p. 1062-1069.

50. Roy, D.K. and L.K. Sharma, Genetic k-means clustering algorithm for mixed numeric and categorical datasets. International Journal of Artificial Intelligence & Applications (IJAIA), 2010. **1**(2).
51. López-Yáñez, I., L. Sheremetov, and C. Yáñez-Márquez, A novel associative model for time series data mining. Pattern Recognition Letters, 2014. **41**: p. 23-33.
52. Lytras, M.D., et al., The Social Media in Academia and EducationResearch R-evolutions and a Paradox: Advanced Next Generation Social Learning Innovation. J. UCS, 2014. **20**(15): p. 1987-1994.
53. Salgado, I., et al., Proportional derivative fuzzy control supplied with second order sliding mode differentiation. Engineering Applications of Artificial Intelligence, 2014. **35**: p. 84-94.
54. Salgado, I., et al., Super-twisting sliding mode differentiation for improving PD controllers performance of second order systems. ISA transactions, 2014. **53**(4): p. 1096-1106.
55. González-Patiño, D., et al., A Novel Bio-Inspired Method for Early Diagnosis of Breast Cancer through Mammographic Image Analysis. Applied Sciences, 2019. **9**(21): p. 4492.
56. Karaboga, D. and B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of Global Optimization, 2007. **39**: p. 459-471.
57. Wilson, R.D. and T.R. Martinez, Improved Heterogeneous Distance Functions. Journal of Artificial Intelligence Research, 1997. **6**: p. 1-34.
58. Brun, M., et al., Model-based evaluation of clustering validation measures. Pattern Recognition, 2007. **40**: p. 807-824.
59. Rosenberg, A. and J. Hirschberg. V-Measure: A conditional entropy-based external cluster evaluation measure. in 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. 2007. Prague.
60. Merz, C.J. and P.M. Murphy, UCI Repository of Machine Learning Databases. University of California at Irvine, Department of Information and Computer Science, Irvine, 1988.
61. Villuendas-Rey, Y. and M.M. Garcia-Lorenzo, Attribute and case selection for nn classifier through rough sets and naturally inspired algorithms. Computación y Sistemas, 2014. **18**(2): p. 295-311.
62. Yáñez-Márquez, C., et al., Emerging computational tools: Impact on engineering education and computer science learning. International Journal of Engineering Education, 2014: p. 533-542.
63. Cerón-Figueroa, S., et al., Instance-based ontology matching for open and distance learning materials. The International Review of Research in Open and Distributed Learning, 2017. **18**(1).
64. García-Floriano, A., et al., Social Web Content Enhancement in a Distance Learning Environment: Intelligent Metadata Generation for Resources. International Review of Research in Open and Distributed Learning, 2017. **18**(1): p. 161-176.

AUTHORS PROFILE

José F. Cabrera-Venegas obtained his B.S. degree in Computer Science from the University of Ciego de Ávila, Cuba, in 2009, and the M.S. degree on Applied Informatics in 2012, from the same institution. He works as a professor of the Computer Science Department of the Faculty of Informatics, University of Ciego de Ávila. He is currently pursuing the Ph.D. degree on Computer Sciences at the Central University of Las Villas, Cuba. His research interests include clustering, bio-inspired algorithms and computational complexity. He is a member of the ACPR (Cuban Association for Patter Recognition) and the Cuban Society of Mathematics, Physics, and Computation.

Yusbel Chávez-Castilla obtained his B.S. degree in Computer Science from the University of Ciego de Ávila, Cuba, in 2009, and the M.S. degree on Applied Informatics in 2012, from the same institution. He works as a professor of the Computer Science Department of the Faculty of Informatics, University of Ciego de Ávila. He is currently pursuing the Ph.D. degree on Computer Sciences at the Central University of Las Villas, Cuba. His research interests include image analysis, clustering, bio-inspired algorithms and computational complexity. He is a member of the ACPR (Cuban Association for Patter Recognition) and the Cuban Society of Mathematics, Physics, and Computation.